Assignment #4

CS432

Timothy Filippone

2/27/2018

Problem 1:

Determine if the friendship paradox holds for my Facebook account.* Compute the mean, standard deviation, and median of the number of friends that my friends have. Create a graph of the number of friends (y-axis) and the friends themselves, sorted by number of friends (y-axis). (The friends don't need to be labeled on the x-axis: just f1, f2, f3, ... fn.) Do include me in the graph and label me accordingly.

Solution:

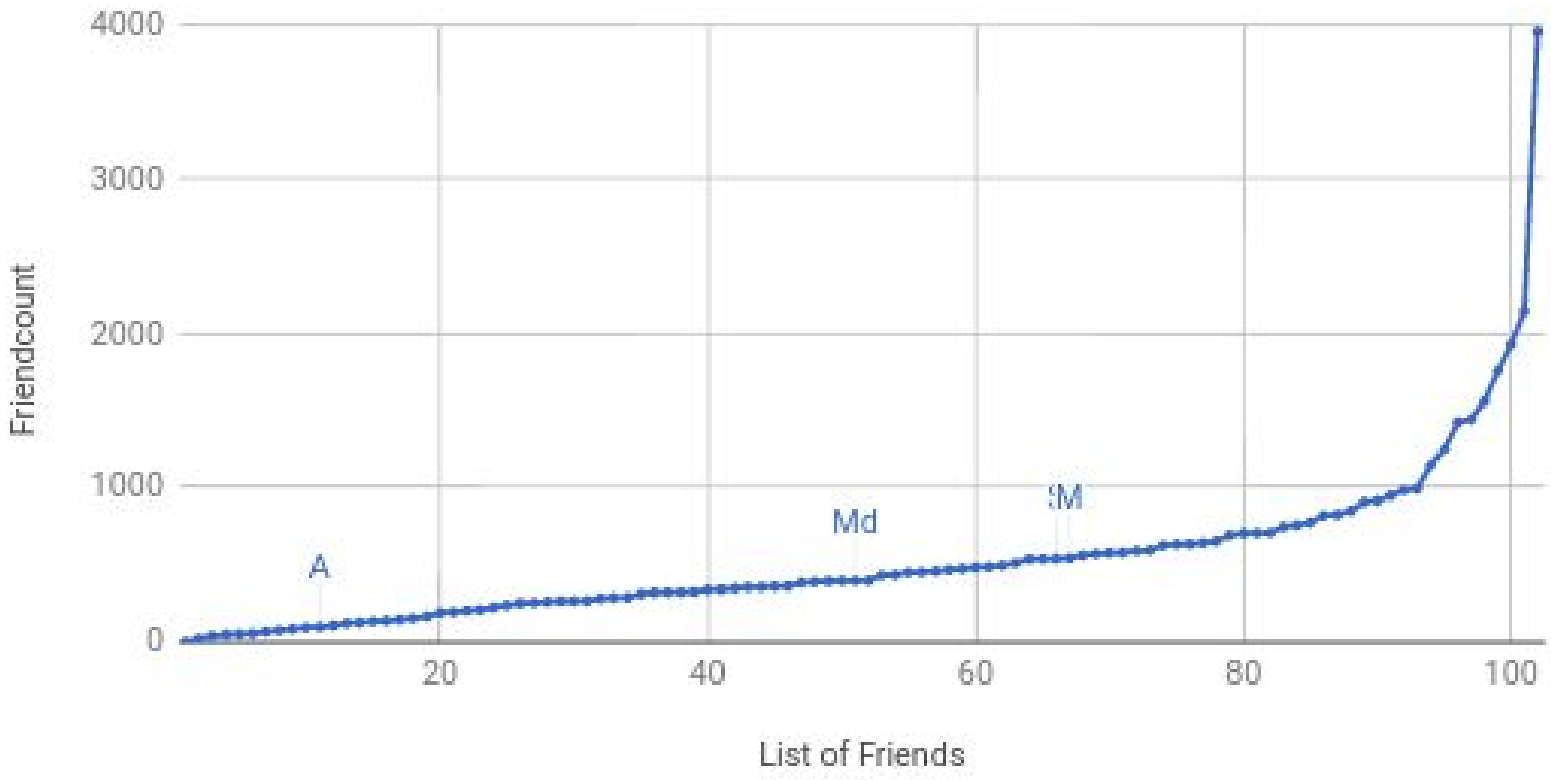
The program `readcsv.py` is responsible for the data results of question 1 in Assignment 4. The library `import pandas` is used to read desired csv files and extract file objects based on predefined dictionary parameters:

```
def readCSV():
    path = "acnwala-friendscount.csv"
    colnames = ["USER", "FRIENDCOUNT"]
    data = pandas.read_csv(path, names=colnames)
    fcount = data.FRIENDCOUNT.tolist()
    users = data.USER.tolist()
...
```

The data is stored in two lists for the user id, along with their friend count. The library `import statistics` is equipped with Mean, Median, and Standard Deviation functionalities that handle the computations for the first two questions.

Facebook List

Legend: A = Anwala(11) Md = Median(51) S = Std Dev(66) M = Mean(67)



Problem 2:

Determine if the friendship paradox holds for your Twitter account. Since Twitter is a directed graph, use "followers" as value you measure (i.e., "do your followers have more followers than you?").

Solution:

Much like Assignment 2, `import tweepy` is heavily used for Twitter's data extraction process. A target user is declared with the `api.get_user` method. An API friendship is created, to authenticate a following between the developer and the target user. Fortunately, as the framework has matured over the years, it is no longer essential to define a Listener class when dealing with these specific cases. Instead, a Cursor object is called in a for loop. The desired api method, in this case followers, along with the user's screen name define the loops condition. From this point, each followers' name and specific follower count can be accessed in each instance.

```
...

for user in tweepy.Cursor(api.followers,
screen_name="acnwala").items():
        follower = api.get_user(user.screen_name)
        print(user.screen_name," ",follower.followers_count)
...
```

Twitter List

Legend: A= Anwala(83) Md= Median(98) M = Mean(178) S = Std Dev(89)